Learning Discretely:
Behaviour and Organisation in Social Learning

Joanna J. Bryson Mark A. Wood

*Artificial models of natural Intelligence (AmonI)
University of Bath, Department of Computer Science
Bath, BA2 7AY, UK
{jjb,cspmaw}@cs.bath.ac.uk

Abstract

This paper describes what is required to learn new tasks in general, then applies this knowledge to understanding imitation learning in specific. We make some reference to the neurological literature, including dual-speed hippocampal / neo-cortical learning systems. We suggest that this model solves the problem of discrete replicants in memetics. We also describe some very preliminary work in implementing and testing our ideas through social learning in a computer game context.

1 Introduction

Human-like intelligence requires an enormous amount of knowledge — solutions to the hard problems of survival and reproduction, which for our species have come to involve complex social and technological manipulations. Some of these solutions are passed to us genetically, and some are learned by an individual during their lifetime through trial-and-error experience. For humans, one key source of knowledge is culture. By *culture* here we mean any knowledge an agent has derived from conspecifics by non-genetic means. In order for such knowledge to be acquired efficiently, the process of acquiring it must be significantly less time consuming (at least for the individual) than individual trial-and-error learning.

In this paper we discuss first how such learning may be accumulated socially by a culture, and then relate this to what we know about learning in individuals. We propose a model for task learning in general, which is clearly facilitated by social information. We then briefly describe our preliminary attempts to build and exploit such a model of learning.

2 Discretion in Memetics

Dawkins (1976b) proposes that knowledge and behaviour can be viewed as developing through a process of evolution, just as biological life has. Ideas or behaviours are propagated if they survive intact long enough to be reproduced. Reproductive success requires replication beyond a single host behaving agent. While some behaviours are known explicitly and transmitted deliberately (by teaching), there is evidence that our species may have evolved the ability to take advantage of this powerful mechanism for increasing knowledge and fitness before we were capable of such explicit mechanisms, and that indeed we still implicitly learn complex multi-modal behaviours from our conspecifics. This allows us to build and transmit knowledge that our cultures have not yet developed words or theories to describe or deliberately represent. This theory of cumulative knowledge generation is called *memetics.*

Dawkins (2000) describes a fundamental problem with the theory of memetics. Memetics is based on the concept of a *meme* which is meant to be analogous to a gene. Some theorists have claimed that this analogy is invalid, on the grounds that genes are discrete, but memes are not. This claim is itself suspect, since to this day the term *gene* still does not describe a well-defined entity (Dennett, 2002), but is based on the fact that the DNA molecule ultimately encodes information in terms of discrete patterns of four possible chemical chains.

The underlying representation for a meme, though still completely unknown, is suspected not to be discrete, and therefore to be open to corruption. To describe the problem, Dawkins (2000) proposes a thought experiment where a child is shown a drawing of an unfamiliar type of boat and asked to copy it; then the process is repeated with another child who sees only the new drawing. Dawkins believes the boat would rapidly become as unrecognisable as a phrase whispered by children playing a game of 'telephone' [Chinese Whispers *U.K.*]. Dawkins proposes a solution to this problem, which is that one learns not gross behaviours, but *instructions* as to how to behave. He proposes an alternate thought experiment, whereby children learn to build a boat by origami, an art based on folding paper. Here small mistakes produced by one child will be corrected by the next, because the second child is able to deduce the intention of the first (or of the designer) because they understand the nature of the operations. In other words, because a process of origami consists of a relatively short list of well-defined operations, Dawkins claims it can be replicated more robustly than a process of drawing.

We believe that Dawkins' requirement that memes must be instructions is over-specific, though correct in principle. We think individuals learn in terms of *skills*, not instructions. There are two differences:

- skills are not necessarily known or communicated explicitly[1], and
- skills are developed by the individual, and thus open to individual variation.

This hypothesis has several interesting ramifications, mostly having to do with the consequences of having variations of *granularity* in memetic representation. For example, consider some teacher J who starts with relatively few mathematical skills, but has by a slow laborious process managed to learn a technique for writing back-propagation networks. Her representation might be a long string of relatively simple arithmetic and trigonometric operators. If she has a student, M, with more mathematical skills (for example, calculus), and he observed her coding a network, he might be able to form a new representation which would create exactly the same sort of system. But M's representation of the system would be quite different from J's, consisting of a smaller number of larger-granularity operators. Note too that the situation could be reversed — if J only knows trigonometry but observes M coding an algorithm, she might well be able to imitate the algorithm herself, however again she would perceive and remember the algorithm at a different level of granularity than that with which M was generating the code.

This sort of model could explain the results of Whiten (2000). Whiten presents various species of primates (including children) with complicated puzzle boxes which require one of a number of sequences of actions to get open. Subjects are generally able to open these boxes if they have first observed a demonstrator, but they will not necessarily go through all the same steps in the same order as the demonstrator. However, if the demonstrator demonstrates repeatedly, on the second or third try the subjects will often perfectly replicate the demonstrator's model, at least in terms of the sequence of affordances used. Subjects may still choose to pull out a pin using their teeth rather than their fingers, for example.

Our explanation would be that initially the subjects are imitating only the goal and perhaps some other simple attributes of the solution (e.g. knowing which knobs on the box need attending to.) However, as they develop skills by opening the box themselves, the difficulty of performing a perfect replication is reduced, because it becomes a relatively short sequence of relatively large-grain actions rather than a long sequence of basic motor commands.

3 Learning in Brains

The hypothesis described above ties in neatly to another hypothesis in learning — this one about how brains can learn from experience.

There are two ways to learn from experience. First, we can learn very slowly, taking a large number of examples to build up a model of how the world seems to be working, or at least what the right thing is to do in a particular context. The second way is to learn very quickly. The problem with learning very quickly is that we may be overly influenced by a very improbable event, taking it to mean more than it should. Learning from a large number of experiences very quickly / perfectly also runs the risk of over-fitting. General-purpose knowledge is usually considered to derive from compiling large amounts of knowledge into a few general rules or policies (Mitchell, 1997, for a summary), although in some relatively deterministic domains it can be derived by extrapolating over a set of exemplars (Poggio, 1990; Atkeson et al., 1997).

Generally speaking, our skills seem to be built up slowly through practise over time. But any such slow-learning system that builds its knowledge from experience faces a problem. The problem is, experience happens quickly. Consequently, what is needed is a second, quick system for jotting down salient events as they happen. McClelland et al. (1995) build a model of such a system, and using the neuroscience literature, tie down their model to particular regions

[1]Though quite probably Dawkins didn't mean to limit memetics to explicit knowledge and was using the term *instruction* in some kind of loose computational metaphor.

of the brain. Slow learning, they say, happens in the neocortex — fast learning happens in the hippocampus (see also Treves and Rolls, 1994)

Another problem with fast learning is that it requires learning a large number of things — particularly if the system needs to hold each learned thing around long enough to allow a slow-learning system to process it. If two different things are learned that happen to be similarly indexed (by whatever category mechanism has emerged in a largely unsupervised system), they might overwrite each other. If accommodation of new information is not done systematically (which is generally seen as the purpose of a slow learning system (McClelland et al., 1995; Mitchell, 1997)) there's no reason to expect two such similarly-indexed events to be neatly, compatibly catalogued together. One way to reduce the probability of such 'collisions' (information about multiple events overwritten into the same locations) is to make sure that the information is encoded in a very sparse way. That is, to use relatively few changes in memory in order to represent the full event. And indeed, this seems to be what the hippocampus does (Rolls, 1996).

In order for a few changes to represent a complex event, each change must be highly salient — it must represent a relatively broad chunk of semantics, a complex concept. As McClelland et al. (1995) point out, this strategy is very compatible with the hippocampal memory indexing theory (Teyler and Discenna, 1986). However, that theory was originally motivated as the use of the hippocampus for a compact, almost symbolic type of representation that would be useful for certain kinds of complex processing. For example, animals without a hippocampus can learn a new map, but they can't learn *how to learn* a map if they've never learned one before (Bannerman et al., 1995). Similarly, animals without a hippocampus can learn to associate actions with stimuli, but they can't learn to prioritise these actions (Alvarado and Bachevalier, 2000; Wood et al., 2004; Buckmaster et al., 2004). Whichever purpose might have originally driven the evolution of a hippocampus, the sparse representation is clearly useful enough to be necessary for at least some sorts of long-term memory storage (Squire et al., 2001), though it's possible that another similar region, perhaps the entorhinal cortex, performs some of the quick-learning roles that McClelland et al. propose for the hippocampus.

We believe that this indexical learning may be based on dynamic categories. That is, the representation of a newly observed behaviour is determined by the 'granularity' of the indexing in the fast-learning system, which is in turn driven by a set of skills learned or formed in the slower learning system. We already know that representations in the hippocampus are highly dynamic and vary by context (Wiener, 1996; Kobayashi et al., 1997). And clearly learned experience is itself a form of context. Thus the hypothesis that what (and how) we can learn with this system changes over time and experience is not excessively radical, although it does have interesting implications for the veracity of recall.

4 A Model of Task Learning

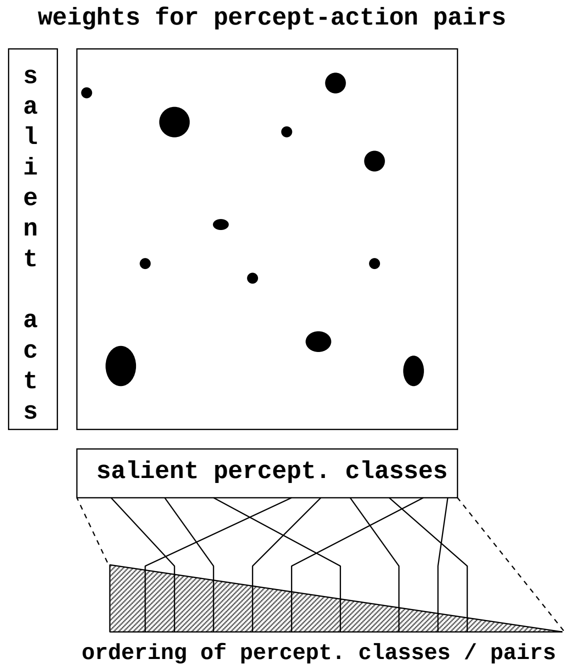

Figure 1: Task learning requires learning four types of things: relevant categories of actions, relevant categories of perceptual contexts, associations between these, and a prioritized ordering of the pairings. Assuming there is no more than one action per perceptual class, ordering the perceptual classes is sufficient to order the pairs. See text for details.

In short, we believe there are at least *four* separate types of things that are learned in the process of learning a task (see Figure 1):

1. *perceptual classes:* What contexts are relevant to selecting appropriate actions.

2. *salient actions:* What sort of actions are likely to solve a problem.

3. *perception/action pairings:* Which actions are appropriate in which salient contexts.

4. *ordering of pairings:* It is possible that more than one salient perceptual class is present at the same time. In this case, an agent needs to know which one is most important to attend to in order to select the next appropriate action.

With respect to perception/action pairings, our current work indicates that there should only be one action possible per salient perceptual context, but there may be many perceptual contexts in which a particular action may be relevant, particularly if the object of the action is coded diectically (Wood et al., 2004; Bryson and Leong, 2005). Also not that although we mention perceptual contexts, we obviously do not mean the full context of all sensory information from a moment in time. Such a representation leads to overfitting / failure to generalize, besides generally being computationally intractable to process. Rather, detailed perception at any particular moment tends to be focussed on a few salient cues which will hopefully help disambiguate the current action-selection problem (Rensink, 2000).

Researchers familiar with Behaviour-Based AI may think of these four sub-problems in a different way. The first three items contribute to forming *behaviour modules* — tight couplings of perception and action, while the last contributes to forming *behaviour arbitration* (Bryson, 2000a; Bryson and Stein, 2001). Researchers familiar with Cognitive Modelling may realise that what we describe is quite similar to ACT-R (Anderson and Matessa, 1998) except with extra emphasis on the forming of categories for sensing and action. However, ACT-R has a relatively simplistic ordering system which cannot account for all animal data on even relatively constrained tasks (Wood et al., 2004). ACT-R learns relatively simple 'utility values' for each perception/action paring, but complex tasks may require hierarchy and/or some other powerful sequence-learning representation such as POMDPs (Kaelbling et al., 1998; Bryson, 2000b).

Clearly solving four problems simultaneously makes learning new skills a very hard problem, but equally it motivates social learning. In a social context, sensing and action categories can be recognised by their co-occurrence (Roy, 1999). In all probability, sequential and hierarchical ordering may also be induced (Dawkins, 1976a).

5 Learning in Practice

In previous work we have shown successful models of solitary primate (including human) task learning where the salient actions and perceptions were already fixed, but the pairings between actions and perceptions and the prioritizations between these varied (Bryson, 2005; Bryson and Leong, 2005). We have also shown that one can create a complete set of possible mappings between perceptual and action classes and then simply prioritize all of these, since only the highest priority item for any perceptual category will be chosen (Wood et al., 2004).

In our current work, we are looking at the role of social learning in perceptual category formation. We are also hoping to explore more complex hierarchical representations. A complete agent needs to be able to move between many different tasks, and indeed determining when one is in a new task context is clearly a part of the problem for determining salient actions and perceptions.

5.1 A Working Model

As in our previous work, we are again not attempting to learn all four categories simultaneously. We have made the following simplifications / assumptions in our preliminary experiments:

- The imitator is initially able to recognise some actions that are key to learning the task.
- Only *one* perceptual class applies to the imitator at any one time.

The second assumption means that, for the time being, we are not worrying about learning priortizations, but merely perceptual classes and their pairings to actions.

Our perceptual classes are defined by boundaries in n-dimensional sensor space (n is the number of sensors providing a reading at any given time). Thus far we have kept n, and the n operative sensors themselves, constant throughout, although having different sets of sensors operating in parallel is one possible way of introducing parallel perceptual classes. The cardinality of each dimension of sensor space differs depending upon the sensor type. For example, a sensor which measures the presence of an object would return a discrete reading $\in \{true, false\}$, whereas a sensor which measures distance would return a continuous reading $\in \mathbb{R}^+$.

The actions we have made recognisable by the imitator are simply discrete. In some sense, actions *must* be discrete (e.g. in categories like turn, move and shout), but they could also be defined by parameters. These in turn can be either absolute (turnTo $north$) or deictic (turnTo $nearest\_actor$) discrete values, but they can also be in terms of continuous values (turn $42.6°$).

Since exactly one perceptual class applies to any given context, and only one action should be associated with any given perceptual class, the problem of perception/action pairing is in this case equivalent to partitioning sensor space and mapping each partition to an action. Given that there is no need for the prioritization of these pairings, this completes our simplified version of the model.

5.2 Domain

For both this initial exploration of perception/action map generation, and future more complex studies, we are carrying out experiments in the domain of virtual-reality computer games. VR games are an excellent platform for experiments involving learning from human subjects because they are real-time, provide a common sensing and action framework for both artificial and human agents, and require many elements of human and animal intelligence, including navigation, reacting to complex, dynamic environments, planning and cooperation (Laird and van Lent, 2001).

We are currently working with two games: *Robocode* (Nelson, 2002) and *Unreal Tournament* (Digital Extremes, 1999).

Robocode is designed to be both a game and a Java teaching tool, provided for free download from IBM alphaWorks. Users have no direct control over their agents, but must provide Java code to drive them. The agents themselves are robotic tanks armed with a single cannon, a few basic sensors, and enough action commands to navigate the map and 'interact' with the other agents therein. The map is a simple 2-D rectangle surrounded by walls, without any obstacles that are not opponents.

Unreal Tournament (UT) is a commercially released, multi-player 'First Person Shooter'. As the term suggests, the user has an agent's-eye view of the game and direct, real-time control of an avatar's actions. UT also supports remote control of agents by sending commands to the game server over a network. This provides a framework for allowing external programs to direct an agents' actions. Such AI-controlled agents are commonly known as 'bots' in the literature and gaming community. The game server, in turn, sends two categories of sensor data back to the client. The first is synchronous: at regular intervals the client is informed of the agent's status (e.g. level of health, amount of ammunition, currently wielded weapon, etc). The other is asynchronous: for example whenever a wall is bumped, a footstep is heard or damage is taken.

5.3 Preliminary Experiments

Our earliest social experiments were conducted in Robocode, because we believed it would be simpler since it was two dimensional (2D) and came with pre-coded sample opponents. However, many aspects of Robocode control and sensing proved inaccessible, presumably to keep competitors from 'cheating' by affecting the code of other robots. Subsequently we have switched to a simple, effectively 2D Unreal Tournament environment.

Our work in UT is still in early stages, but we have had agents successfully learn simple plans from observation. In addition to providing a basic proof of concept, these experiments also point to representational issues which lie ahead. These will be discussed below.

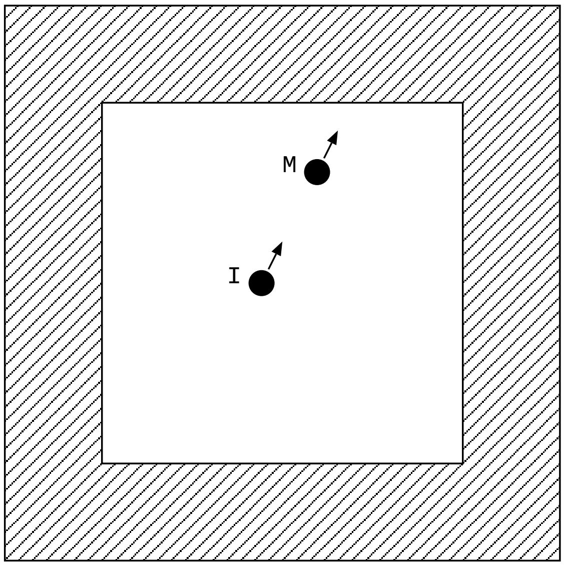

Figure 2: The experimental arena

The experiment consisted of two actors (bots) moving within a single cuboid room (see Figure 2). World co-ordinates are given in three dimensions by the game engine, but since the bots only moved on the floor plane, the problem is well-defined in two dimensions. Similarly, the bots have three rotational degrees of freedom, but only one is used here (2D heading).

The **model** bot (labelled *M* in the figure) executes the following behaviour: move forward if not too close to a wall; otherwise turn away from the nearest wall and then move forward. The actual distance at which the proximity sensor is triggered is determined by a setting in the sensor module. The region that this state applies to is represented by the shaded area in the figure. The angle through which the bot turns is calculated randomly, constrained by the fact that the bot must then head away from the nearest wall.

The goal of the **imitator** bot (labelled *I* in the figure) is to locate a model, and then remain a fixed distance behind it and record observations (after Billard and Dautenhahn (2000)). In this toy environment, it probably would have been sufficient to have a stationary imitator, but for larger and more complex environments and model behaviours, the imitator would need to stay close to its model in order to observe as closely as possible what the model observes. The imitator needs to be aware of when the model initiates a new action, so that it can record the sensor state at that instant and use it later to construct a perception/action mapping (see Section 5.1). We have tried two types of cue for this purpose:

1. The model acts explicitly as a teacher, informing the imitator of its decisions as and when they are made. The imitator only records an observation when this cue is given.

2. The model is passive, forcing the imitator to take snapshots of the sensor space at some predetermined regular interval.

The former simulates the training of a team-mate, i.e. where the goal of the model is for the imitator to learn as efficiently as possible. This method could not, however, be used to learn behaviour from 'unhelpful' agents (such as opponents). The latter could be used in this way, but risks missing the decision instant if the observation frequency is too low. There is also a risk of storing redundant data if thresholds between motions are not accurately detected. Nevertheless, either of these problems should be addressable given sufficient learning opportunities and a robust probabilistic representation.

Whichever cue is used, we endow the imitator with the ability to recognise the actions *move forward* and *turn*. The first set of sensors we gave to the imitator detected the x- and y-position respectively of the model in World co-ordinates, resulting in a 2D sensor space. At first glance, the partition would seem to be obvious; in fact directly analogous to the plan shown in Figure 2. The problem is that *move forward* decisions are taken both in the white zone, and in the shaded zone immediately after the robot has finished turning. If the shaded area cannot be mapped to a unique action, then the partition it generates is unsuitable. In fact, there is no suitable partition of this sensor space. Even if we take a more powerful representation and give the imitator a sensor that detects the distance of the model from the nearest wall, the problem remains.

There are (at least) two ways to solve this problem. The first is to give the imitator a sensor which detects the past (commonly known as memory) or, more specifically, detects the previously recorded action. If we use this in tandem with the distance sensor, we can create the following map: if close to a wall and the previously recorded action was *move forward* then *turn*; otherwise *move forward*. This makes sense, as there is an implicit two-item sequence present in the behaviour of the model. The second is to add to the distance sensor another which can detect whether or not the model is facing the nearest wall. The resulting map is equivalent to the one above: if close to a wall and facing it then *turn*; otherwise *move forward*. This also makes sense, as the model's behaviour contains a piece of state indicating whether or not it is facing the nearest wall. What is noteworthy is the two different ways of solving the same problem; one temporal and one atemporal.

5.4 Discussion and Future Work

Given this ambiguity, our next task is to investigate whether harder tasks are better solved by a greater number of 'immediate' sensors, or by the introduction of temporal dependencies. We expect POMDPs (Kaelbling et al., 1998), which we also mentioned in Section 4, will provide a way of more naturally modelling temporal systems, as well as the latent variables which are bound to be components of more complex behaviour.

Also, as we alluded to in Section 5.1, we conjecture that grouping sensors into modules that compete probabilistically for saliency in a particular action context will create naturally competing perceptual classes which will in turn need prioritization (see part 4 of the task learning model in Section 4). Using the scenario in Section 5.3, as an example suppose we created several sensor modules each containing one sensor as follows:

1. Distance of model from nearest wall.

2. Distance of model from second nearest wall.

3. Distance from the North wall.

After repeated observations it would become clear that module 1 influences the decision process of the model with a far greater probability than modules 2 or 3; that perceptual class should be given a higher priority at least in the context of generating turns. On the other hand, if there is a door in the North wall, for some other tasks the absolute location may be more salient. In general, we expect an agent will need to actively maintain modules which utilise different viewpoints (e.g. World (absolute / allocentric) view, model / imitator (egocentric) view, teammate- and opponent-oriented views, etc.) to see which provide the most easily interpretable behaviour data in different contexts.

Currently however we are still working with relatively simple representational issues, including that of discreteness. Many actions are not easy to discretize: a bot that is turning may make one decision to turn in a long, continuous arc, or many consecutive decisions to turn in a series of smaller arcs. As we said in Section 2, it may not be important that the imitator forms the same perceptual categories as the model is using. In particular, since our models are using a radically different action-selection mechanism than our imitators, it is actually quite likely that the optimal behavioural categories may be different.

To reiterate our hypothesis, we assume that, some of these discriminations will be informed by skills the learning agent has already accumulated, whether through individual learning, previous imitation learning, or by 'innate' predisposition.

6 Summary

If we can build a model of task learning in the games domain, then it will be fairly simple to test how much social learning of a task can accelerate task learning by individual agents, as we can easily create experimental subjects that do or don't attend to other agents in the room. Also, if we allow for both individual and social learning at the same time, we believe we will quite naturally demonstrate agents with similar expressed behaviour, but with different internal representations. Finally, assuming we acquire learners with different perceptual categories (either through learning as just described or by programming) we will be able to test in what circumstances successful behaviours can be propagated through multiple 'generations' across multiple learning agents.

In this paper, we have described one of the key problems in memetics, the problem of discreteness in the representation of behaviour observed in conspecifics. We have suggested that the units of memetics may in fact vary between individuals based on their skills, knowledge and random factors in the self-organization of the underlying neurological representations of these. This may not be a bad thing, in fact it may account for why some observers are able to exceed the performance of their models. We have proposed a framework for representing this sort of learning and described preliminary experiments in building and using such a framework for social learning in the context of real-time multi-player computer games. In the future, we hope to radically expand our experiments and in the process continue to refine our model.

References

Maria C. Alvarado and Jocelyne Bachevalier. Revisiting the maturation of medial temporal lobe memory functions in primates. *Learning and Memory*, 7(5):244–256, 2000.

J. R. Anderson and M. Matessa. The rational analysis of categorization and the ACT-R architecture. In M. Oaksford and N. Chater, editors, *Rational Models of Cognition*. Oxford University Press, 1998.

Christopher G. Atkeson, Andrwe W. Moore, and Stefan Schaal. Locally weighted learning for control. *Artificial Intelligence Review*, 11:75–113, 1997.

D. M. Bannerman, M. A. Good, S. P. Butcher, M. Ramsay, and R. G. M. Morris. Distinct components of spatial learning revealed by prior training and NDMA receptor blockade. *Nature*, 378: 182–186, 1995.

Aude Billard and Kerstin Dautenhahn. Experiments in social robotics: grounding and use of communication in autonomous agents. *Adaptive Behavior*, 7(3/4), 2000.

Joanna J. Bryson. Cross-paradigm analysis of autonomous agent architecture. *Journal of Experimental and Theoretical Artificial Intelligence*, 12 (2):165–190, 2000a.

Joanna J. Bryson. Hierarchy and sequence vs. full parallelism in reactive action selection architectures. In *From Animals to Animats 6 (SAB00)*, pages 147–156, Cambridge, MA, 2000b. MIT Press.

Joanna J. Bryson. Evidence of modularity from primate errors during task learning. In *Modelling Language, Cognition, and Action: The Nineth*

Neural Computation and Psychology Workshop (NCPW9), 2005.

Joanna J. Bryson and Jonathan C. S. Leong. Primate errors in transitive 'inference': Representations underlying transitive choice in humans and other primates. submitted for review, 2005.

Joanna J. Bryson and Lynn Andrea Stein. Modularity and design in reactive intelligence. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, pages 1115–1120, Seattle, August 2001. Morgan Kaufmann.

Cindy A. Buckmaster, Howard Eichenbaum, David G. Amaral, Wendy A. Suzuki, and Peter R. Rapp. Entorhinal cortex lesions disrupt the relational organization of memory in monkeys. *The Journal of Neuroscience*, 24(44):9811–9825, November 3 2004.

Richard Dawkins. Hierarchical organisation: A candidate principle for ethology. In P. P. G. Bateson and R. A. Hinde, editors, *Growing Points in Ethology*, pages 7–54. Cambridge University Press, Cambridge, 1976a.

Richard Dawkins. *The Selfish Gene*. Oxford University Press, 1976b.

Richard Dawkins. Foreword to Susan Blackmore's *Meme Machine*, pages *vii–xvii*. Oxford University Press, second edition, 2000.

Daniel C. Dennett. The new replicators. In Mark Pagel, editor, *The Encyclopedia of Evolution*, volume 1, pages E83–E92. Oxford University Press, Oxford, 2002.

Digital Extremes. *Unreal Tournament*, 1999. Epic Games, Inc.

Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1–2):99–134, 1998.

T. Kobayashi, H. Nishijo, M. Fukuda, J. Bures, and T. Ono. Task-dependent representations in rat hippocampal place neurons. *JOURNAL OF NEUROPHYSIOLOGY*, 78(2):597–613, 1997.

John E. Laird and Michale van Lent. Human-levelAI's killer application: Interactive computer games. *AI Magazine*, 22(2):15–25, 2001.

James L. McClelland, Bruce L. McNaughton, and Randall C. O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3):419–457, 1995.

Tom Mitchell. *Machine Learning*. McGraw Hill, New York, NY, 1997.

Mathew Nelson. *Robocode*, 2002. IBM alphaWorks.

Tomaso Poggio. A theory of how the brain might work. Memo 1253, Massachusetts Institute of TechnologyArtificial Intelligence Lab, Cambridge, MA, December 1990.

Ronald A. Rensink. The dynamic representation of scenes. *Visual Cognition*, 7:17–42, 2000.

Edmund T. Rolls. A theory of hippocampal function in memory. *Hippocampus*, 6(6):601–620, 1996.

Deb Kumar Roy. *Learning from Sights and Sounds: A Computational Model*. PhD thesis, MIT, Media Laboratory, sep 1999.

Larry R. Squire, Robert E. Clark, and Barbara J. Knowlton. Retrograde amnesia. *Hippocampus*, 11 (1):61–71, 2001.

T. J. Teyler and P. Discenna. The hippocampal memory indexing theory. *Behavioral Neuroscience*, 100:147–154, 1986.

Alessandro Treves and Edmund T. Rolls. A computational analysis of the role of the hippocampus in memory. *Hippocampus*, 4:374–391, 1994.

Andrew Whiten. Primate culture and social learning. *Cognitive Science*, 24(3):477–508, 2000.

S. I. Wiener. Spatial, behavioral and sensory correlates of hippocampal CA1 complex spike cell activity: Implications for information processing functions. *Progress in Neurobiology*, 49(4):335, 1996.

Mark A. Wood, Jonathan C. S. Leong, and Joanna J. Bryson. ACT-R is *almost* a model of primate task learning: Experiments in modelling transitive inference. In *The* 26^{th} *Annual Meeting of the Cognitive Science Society (CogSci 2004)*, pages 1470–1475, Chicago, 2004. Lawrence Erlbaum Associates.